REMARKS

Applicant expresses appreciation to the Examiner for consideration of the subject patent application. This amendment is in response to the Office Action mailed August 28, 2006. Claims 10-14 were rejected. The claims have been amended to address the concerns raised by the Examiner.

Claims 1-23 were originally presented. Claims 10-14 remain in the application. Claims 1-9 and 15-23 were previously cancelled without prejudice. No claims have been amended. Claims 24 - 38 have been added.

Claims 1-9 and 15-23 were previously cancelled to put the application in a condition for allowance of allowable subject matter. However, the allowance has since been withdrawn. Therefore, claims 1-9 and 15-23 have been reinstated as claims 24-38 (with initial claims 2, 16, and 21 having been previously cancelled and their subject matter added to the independent claims). No new matter has been added.

Claim Rejections - 35 U.S.C. § 102

Claims 10, 11, and 13 (including independent claim 10) were rejected under 35 U.S.C. § 102(e) as being anticipated by Kajiya et al., US Patent No. 5,864,342 (hereinafter referred to as "Kajiya"). In order to explain the differences between the prior art and the current claims, a description of Kajiya will be provided and then followed by a discussion of how the current claims are distinguished over the prior art.

Kajiya discloses a method for rendering graphical objects in a scene to generate a display image by dividing the geometric primitives of models in the scene into chunks of the view space to which the primitives will be rendered. Pixel fragments can then be resolved in a post-processing step for one chunk while primitives for another chunk are rasterized. (See Abstract)

For example, FIG. 4A of Kajiya discloses a block diagram showing the process for rendering and displaying the scene (See Col. 13, line 54 to Col. 14, line 41). A DSP 176 is used to perform pre-processing of image data, such as video compression/decompression and front-end graphics processing (transformations, lighting, etc.).

The DSP 176 outputs the pre-processed and geometrically transformed image data to a rendering engine referred to as a tiler 200. The tiler performs scan-conversion, shading, texturing, hidden-surface removal, anti-aliasing, translucency, shadowing, and blending for multi-pass rendering. **This means that the rendering is fully complete before any operation in the gsprite engine takes place.** The resulting rendered "gsprite" chunks are then compressed and stored in compressed form in a shared memory. After the gsprite chunks are rendered and stored, several other operations are necessary before the rendered chunks can be displayed.

The output of the tiler 200 rendering engine is sent to a gsprite engine 204 that is used to decompress gsprite chunk data and perform the necessary image processing for general affine transformations, including scaling, translation with subpixel accuracy, rotation, reflection, and shearing.

After filtering, the resulting pixels (with alpha) are sent to the compositing (post processing) buffers where display pixel data is calculated. The Office Action infers that post processing is equivalent to rendering and displaying. However, this is not accurate. A post processor provides the functionality for doubled buffered overlays, landing light lobes, and edge blending attenuation using the alpha memory. The invention disclosed in Kajiya is rendering using a double buffer system and the rendering for each frame is completed before the frame reaches the post processor 210 and DAC 212 display systems.

Gsprite chunk data is processed a number of scan lines at a time for display. Chunk data is processed 32 scan lines at a time. The output of the Gsprite engine is sent to the compositing buffer 210, a post processing buffer that includes two 32 scan line color buffers which are toggled between display and compositing activities. The compositing buffer also includes a 32 scan line alpha buffer which is used to accumulate alpha for each pixel. The output of the post processing buffer is sent to a DAC 212 and a display screen. While the gsprite engine 204 fills a compositing buffer for one band, the compositing buffer transfers composited image data for another band to the DAC 212. In the next band period, the band that was just composited is then displayed. This process repeats for bands in the display. **Because of this double-buffering**, the process of transforming and compositing of pixels can occur simultaneously with the process of displaying a band. (see Col. 60, line 63 to Col. 61, line 13). **The two buffers ping-pong back and forth so that as one scanline region is being displayed, the next is being composited.**

In contrast to the method for rendering graphical objects disclosed in Kajiya, the present application recites in claim 10 a method for enabling **a single pixel frame buffer** for simultaneous rendering and display in a computer image generator. The method includes the steps of:

(c) rendering the screen bins by row from top to bottom, into the pixel frame buffer; and

(d) **displaying at least one row of screen bins rendered before the rendering of all the screen bins has completed**, wherein the displaying of the screen bins takes place after a selected portion of the screen bins for a current field have been rendered.

For example, the application discloses that once the geometry buffer between the geometry and rendering processes toggles, the bins can be rendered in sequential order. Once the top row of bins has been rendered into the pixel frame buffer, they can be displayed. (See App. page 11, lines 3-11).

Kajiya does not teach or suggest displaying at least one row of screen bins rendered before rendering of all the screen bins has been completed, as recited in independent claim 10. Rather, as previously discussed, Kajiya renders a scene using a tiler rendering engine 200, compresses the results, and stores the results in a memory 216. The stored results then undergo a geometric transformation 204 followed by post-processing 210 before being sent to a display.

The Office Action cites column 6, lines 15-29; column 60, lines 24-47 and 63-67; and column 61, lines 11-13 as teaching the step of displaying at least one row of screen bins after at least one row has completed rendering. However, Kajiya teaches that the step of **displaying is completed after each frame of the scene has been rendered, compressed, and stored**. The two compositing buffers disclosed in column 61 occur in the post-processor 210 after the rendering engine 200 and geometric transformation engine 204. **Thus, Kajiya does not disclose displaying at least one row of screen bins rendered before rendering of all the screen bins has been completed**, as recited in claim 10. Rather, as taught in the abstract, Kajiya teaches that pixel-fragments can be resolved **in a post-processing step** for one chunk while primitives for another chunk are rasterized.

Therefore, Applicant respectfully submits that independent claim 10 is allowable, and urges the Examiner to withdraw the rejection.

Rejection of the dependent claims 11 and 13 should be reconsidered and withdrawn for at least the reasons given above with respect to the independent claim. The dependent claim, being narrower in scope, is allowable for at least the reasons for which the independent claim is allowable.

Further, Kajiya teaches double buffering even though the double buffers are smaller than conventional double buffers. This means that Kajiya teaches the use of a pixel frame buffer in combination with a smaller double buffered rendering area. This increases the frame buffer memory used by Kajiya by two screen band buffers or two chunk buffers, whereas the present invention uses only the single frame buffer.

The present invention does not use double buffering. Claim 24 (previously claim 1) includes the limitation of "enabling **a single pixel frame buffer** for simultaneous rendering and display in a computer image generator." This limitation is also included in paragraph (d) of claim 1. The prior art does not teach or suggest using a single pixel frame buffer.

Applicants teach subdividing a screen into screen bins to allow a single pixel frame buffer to simultaneously receive and send data for display (Application claim 1; page 7, lines 18-19). This is significantly different than swapping buffers because a hardware interlock is not required to be used. The simultaneous receiving and sending of data of Applicant's method included in claim 1 includes the step of "displaying at least one rendered screen bin before the rendering of all the screen bins has completed for the single pixel frame buffer." Kajiya must complete rendering for an entire buffer the buffer data can then be output.

Kajiya's remedy for overload requires aggregating objects to reduce the processing time for a band (Kajiya col 60, lines 21-28). Kajiya does not teach simultaneously receiving and sending data for display to a frame buffer, but instead requires a double screen band buffer that alternates receiving and sending data every band. The single buffer of the present invention does not switch like a double buffer and does not need double buffering because the delay time between rendering and displaying of bins can be modified.

Each screen bin is rendered and stored in the pixel frame buffer. After at least one screen bin row has been rendered, the buffer may be sent to the display (Application claim 1; page 11, lines 5-14). Since the entire pixel frame buffer is present, the remedy for overload is a change in the delay from display start to rendering start (Application figures 7-9 showing adjustment timing; figures 10-11 showing overload timing; page 13, lines 3-7).

The Office Action states that Kajiya teaches a method for displaying at least one rendered screen bin before the rendering of all the screen bins has completed for the single pixel buffer (Office Action Page 3). However, Kajiya does not use a single pixel buffer for their display method, but instead uses a double buffer sized to store a frame band or a chunk. Kajiya must render an entire buffer before it can switch buffers for output. The present invention allows part of a frame buffer to be output before rendering for the buffer is complete.

Kajiya's display method provides more memory for double buffers with frame bands (Col. 60, lines 62-67), while Applicant claims an entire single pixel frame buffer without additional double buffering. Kajiya's double buffer can only receive or send data during a band cycle (Col. 60, lines 63-66), while Applicant's system uses an entire single pixel frame buffer (Application Claim 24) that can simultaneously receive and send data (Application Claim 24).

Since Kajiya's method limits writing to the band that is not being displayed, the next band is displayed right after the current band finishes (Col. 60, lines 63-67). In contrast, the single pixel frame buffer allows the Applicant's solution to adjust the timing of rendering start versus display start (Application figures 7-9 showing adjustment timing; figures 10-11 showing overload timing; page 12, lines 18-19). Kajiya's solution to overloading is to aggregate objects (Col. 60, lines 26-28). Kajiya does not disclose a remedy for overloading by controlling of the start of rendering with respect to the start of displaying.

The Office Action states that Kajiya shows the step of displaying at least one rendered chunk before the rendering of all the chunks has completed from a buffer (Office Action Page 3). However, Kajiya does not show displaying at least one rendered chunk in a buffer before rendering of all the chunks in the buffer, but requires rendering an entire screen band buffer before that buffer is displayed (Col. 60, lines 59-67).

Kajiya cannot render more or less than an entire buffer with his method of double buffering because only one band or chunk is being composited at a time (Col. 60, lines 63-66).

Applicant's method does not require an entire buffer to be filled before the buffer is displayed. Applicant's invention involves the adjustment of timing the number of screen bins that are rendered before display (Application claim 24; Application page 12, lines 18-19).

A previous Office Action stated that Kajiya does disclose the idea of single buffering and cited Kajiya, Col. 6, lines 15-19. However, this reference in the summary of the patent is obscure and difficult to understand. The reference describes a "single rasterization buffer". In the next paragraph, it is explained that the rasterization buffer is a double buffer. The specification does not teach or suggest the idea of using a single pixel frame buffer, but describes on several occasions the use of a double-buffering system. Therefore, it appears that the correct interpretation of the cited reference in the patent summary (Col. 6, lines 15-19) is that the system in Kajiya can use a single, double-buffered rasterization buffer.

Since Kajiya does not show or teach a single pixel frame buffer, a variable number of completed screen bins before display, or a remedy for overload involving distributed processing time Kajiya et al. does not anticipate applicants' claim 24, 33, and 41, and the claims depending therefrom.

Claim Rejections - 35 U.S.C. § 103

Claims 12 and 14 were rejected under 35 U.S.C. § 103 as being unpatentable over Kajiya in view of Grigor et al. (hereinafter "Grigor"), US Patent No. 6,853,381.

Rejection of the dependent claims 12 and 14 should be reconsidered and withdrawn for at least the reasons given above with respect to the independent claims. The dependent claims, being narrower in scope, are allowable for at least the reasons for which the independent claims are allowable.

CONCLUSION

In light of the above, Applicant respectfully submits that pending claims 10-14, 24, 26-33, and 35-41 are now in condition for allowance. Therefore, Applicant requests that the rejections and objections be withdrawn, and that the claims be allowed and passed to issue. If any impediment to the allowance of these claims remains after entry of this Amendment, the Examiner is strongly encouraged to call Steve Perry at (801) 566-6633 so that such matters may be resolved as expeditiously as possible.

Fifteen claims were added (claims 24-38), including 3 independent claims (claims 24, 32, and 36), while seventeen claims were canceled (claims 1-9 and 15-23), including three independent claims (claims 1, 15, and 20). Therefore, no additional fee is due.

The Commissioner is hereby authorized to charge any additional fee or to credit any overpayment in connection with this Amendment to Deposit Account No. 20-0100.

DATED this 28th day of November, 2006.

Respectfully submitted,

/Steve M. Perry/

Steve M. Perry
Registration No. 45,357

THORPE NORTH & WESTERN, LLP
Customer No. 20,551
P.O. Box 1219
Sandy, Utah 84091-1219
Telephone: (801) 566-6633